



**HAL**  
open science

## Distributed FDI of a networked embedded microdrone

Aneel Tanwani, Sylviane Gentil, Suzanne Leseq, Jean-Marc Thiriet

► **To cite this version:**

Aneel Tanwani, Sylviane Gentil, Suzanne Leseq, Jean-Marc Thiriet. Distributed FDI of a networked embedded microdrone. Workshop on Advanced Control and Diagnosis, ACD'2006, Nov 2006, nancy, France. 6 p. hal-00125498

**HAL Id: hal-00125498**

**<https://hal.science/hal-00125498>**

Submitted on 19 Jan 2007

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## DISTRIBUTED FDI OF A NETWORKED EMBEDDED MICRODRONE

A. Tanwani, S. Gentil, S. Leseq, J-M. Thiriet

*Laboratoire d'Automatique de Grenoble (UMR 5528 CNRS, INPG, UJF)  
BP 46, 38402 St Martin d'Hères France  
tel : (33) 476-82-62-39 - fax : (33) 476-82-63-88  
{Suzanne.Leseq,Sylviane.Gentil,Jean-Marc.Thiriet}@inpg.fr  
tanwani2@uiuc.edu*

Abstract: Embedded systems constitute a category whose safety is critical and where FDI real time constraints are particularly important. Embedded algorithms must be the simplest possible and computations may be distributed between the embedded system and a more powerful distant computer. This paper proposes a bank of observers to diagnose faults of a small helicopter controlled in closed loop. The studied prototype is a 4 rotors mini drone equipped with an attitude central for system positioning, made of 3 gyroscopes, 3 accelerometers and 3 magnetometers. A dedicated network architecture is proposed.

Keywords: Fault Diagnosis, Embedded system, Generalized Observer Scheme Residual Generation, Distributed computer control system, Network Topology, Drones.

### 1. INTRODUCTION

Drones constitute a category of embedded systems whose safety is critical. The system must still be able to exhibit given properties, such as stability or precision, even in the presence of faults. Compared to aircrafts, they constitute low cost systems whose sensors and actuators are often subject to faults (offsets, drifts). Consequently, their maintenance is essential and difficult. Hardware redundancy is not conceivable, not only due to cost but also to constraints on weight, energy and operating space.

*Fault Detection and Isolation (FDI) real time constraints are particularly important for these systems.* Early detection is necessary but embedded algorithms must be as simpler as possible. Thus computations may be distributed between the embedded system and a more powerful distant computer sharing information through a network. Control, state estimation, fault detection, fault isolation and controller accommodation or reconfiguration must be considered as cooperative functions distributed on the network architecture. Splitting of algorithms is an open problem.

NCS emerged recently with the rapid development of wire or wireless network technology. It is now reasonable to design at the same time the network and the control and FDI algorithms (Nilsson et al, 1998; Otanez et al., 2002). Evidently, from the FDI point of view the design of residuals and their evaluation is more difficult (Fang et al., 2006). Some faults are induced by the network, such as packet losses comparable to an intermittent sensor failure. Delays are induced by the network and should be taken into account by the algorithms (Zhang and Ding, 2006).

This paper deals with a particular application. This preliminary work is completed in simulation. Section 2 presents the studied prototype: a 4 rotors mini drone equipped with an attitude central for system positioning, made of 3 gyroscopes, 3 accelerometers and 3 magnetometers. The system is controlled in closed loop through a state observer. Several kinds of realistic faults can be hypothesized on this system, whose study is proposed in (Tanwani et al., submitted). The following sections focus on the case of sensor and actuator faults. In section 3, a model-based diagnostic approach is adopted. A model of the system has been designed for the control synthesis, simulated with SIMULINK and extended to fault analysis. An FDI algorithm based on a generalized

observer scheme is proposed and results are commented. Section 4 proposes a network architecture to embed or remotely distribute control and FDI algorithms.

## 2. PRESENTATION OF THE SYSTEM

A drone with 4 rotors is regarded as a composition of two PVTOL (Planar Vertical Take off Landing) problems whose axes are orthogonal allowing a movement of six degrees of freedom.

### 2.1 Mechanical model

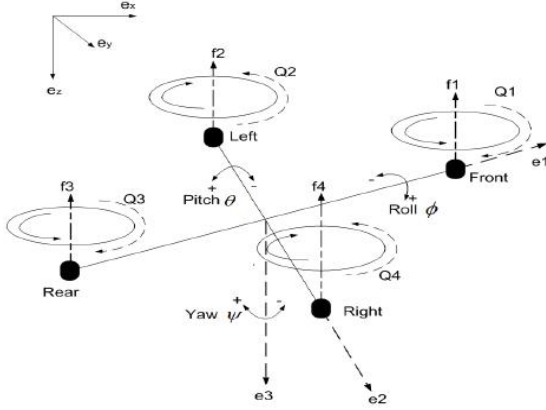


Fig. 1. Coordinate frames assigned to the drone

The four electric motors, driving the four blades, provide the thrust for the movement of the drone (Tayebi and Mc Gilvray, 2004), Fig. 1. To develop the dynamic equations, consider R, an inertial frame of reference and B, a frame attached to the drone. The dynamic model is described by the following equations:

$$\dot{p} = v, \quad (1)$$

$$\dot{v} = g e_z - \frac{1}{m} T R_B^R e_z \quad (2)$$

$$\dot{q} = \frac{1}{2} q \otimes \Omega = \frac{1}{2} \Omega(\varpi) q \quad (3)$$

$$I_f \dot{\varpi} = -\varpi \times I_f \varpi - G_a + \tau_a \quad (4)$$

$$I_r \dot{\omega}_i = \tau_i - Q_i, i \in \{1, 2, 3, 4\} \quad (5)$$

where  $m$  indicates the mass of the drone,  $g$  is acceleration due to gravity,  $e_z = (0, 0, 1)^T$  is a unit vector in reference frame R, the vector  $p = (x, y, z)^T$  indicates the position of the origin of frame B relative to frame R, the vector  $v = (v_x, v_y, v_z)^T$  is the linear velocity of the origin of B measured in the reference frame R. The orientation of the drone is given by the orthogonal rotation matrix  $R_B^R \in SO(3)$ .  $I_f \in \mathfrak{R}^{3 \times 3}$  is a symmetric positive definite constant inertia matrix of the drone with respect to the frame R whose origin is at the centre of mass. The speed and the moment of inertia of rotor  $i$  are indicated, respectively, by  $\omega_i$  and  $I_r$ . The reactive torque generated by rotor  $i$  is given by:

$$Q_i = k \omega_i^2 \quad (6)$$

and the total thrust generated by four rotors is given by:

$$T = \sum_{i=1}^4 f_i = b \sum_{i=1}^4 \omega_i^2 \quad (7)$$

$k > 0$  and  $b > 0$  are two parameters depending on the air density, the radius, the shape, the pitch angle of the blade and other factors. The gyroscopic torques, due to the combination of the rotation of the drone and the four rotors, are modelled as:

$$G_a = \sum_{i=1}^4 I_r (\varpi \times e_z) (-1)^{i+1} \omega_i \quad (8)$$

The drone torques generated by rotors are given by:

$$\begin{aligned} \tau_a^1 &= db(\omega_2^2 - \omega_4^2) \\ \tau_a^2 &= db(\omega_1^2 - \omega_3^2) \end{aligned} \quad (9)$$

$$\tau_a^3 = k(\omega_1^2 + \omega_3^2 - \omega_2^2 - \omega_4^2)$$

where  $d$  represents the distance from the rotors to the centre of mass of the drone, and  $\tau_i$  represents the torque produced by rotor  $i$ .

In equation (3), the attitude dynamics are represented by  $q = (q_0, \bar{q})^T$  which is a rotational quaternion (Chou, 1992; Guerrero et al. 2005) that gives the orientation of the drone.  $\Omega = (0, \varpi)^T$  is a quaternion with  $\varpi$  the angular velocity of the drone measured by the gyrometers in the frame B.  $\otimes$  indicates quaternion multiplication which leads to the following definition of  $\Omega(\varpi)$ :

$$\Omega(\varpi) = \begin{pmatrix} 0 & -\varpi^T \\ \varpi & -[\varpi^\times] \end{pmatrix} \quad (10)$$

where  $[\varpi^\times]$  is the self cross product.

### 2.2. Sensors

The estimation of attitude (or orientation), position and speed is the pre requisite for flight control. For this purpose, a central navigation unit is used that consists of a tri-axis accelerometer ( $a_1, a_2, a_3$ ) a tri-axis magnetometer ( $m_1, m_2, m_3$ ) and a tri-axis rate gyroscope ( $g_1, g_2, g_3$ ). These sensors use MEMS (Micro Electronic and Mechanical Systems) technology to reduce the weight, dimension and cost.

1) **Rate Gyros:** The angular velocity  $\varpi$  is measured by tri-axis rate gyros that are usually affected by various errors, such as fixed bias, drift bias, fixed scale or asymmetric scale factor error, and noise. Theoretically, the integral of  $\varpi$  gives the relative orientation but the presence of bias at low frequencies generates errors which accumulate over time. The sensor measurements are modelled as:

$$\dot{\varpi} = \varpi_g - v - \eta_1, \dot{v} = -\frac{1}{\tau} v + \eta_2 \quad (11)$$

where  $\varpi_g$  is the sensor value,  $v$  is the bias,  $\tau = 100$  s is the time constant obtained from an Allan Variance chart, and  $\eta_i$  are assumed to be Gaussian white noises.

2) **Accelerometers:** The 3-axis accelerometer senses inertial forces and gravity in the aircraft body coordinate system B. The transformation of force and gravity from inertial frame R to body frame is given by:

$$\vec{b}_{acc} = C(q)(\vec{a} - g\vec{e}_z) + \eta_{acc} \quad (12)$$

where  $\vec{b}_{acc}$  is the value measured by sensors. The motion is supposed quasi static: linear accelerations are negligible. In this way, accelerometers are only sensible to earth's gravitational pull.

The rotation matrix  $C(q)$  is the Rodrigues matrix, defined as:

$$C(q) = (q_o^2 - \vec{q}^T \vec{q})I + 2(\vec{q}\vec{q}^T - q_o [q^\times]) \quad (13)$$

3) **Magnetometers:** the information provided by a 3-axis magnetometer is added to the inertial measurements. The magnetic field vector is transformed to the mobile frame attached to the aircraft and is defined by:

$$\vec{b}_{mag} = C(q)\vec{h}_m + \eta_{mag} \quad (14)$$

where  $\vec{h}_m = (\vec{h}_m \ 0 \ \vec{h}_m)^T$  and  $\vec{b}_{mag}$  are the components of the magnetic field vector in R and B, respectively.

### 3. STATE OBSERVER AND CLOSED LOOP CONTROL

The measurements provided by the above mentioned accelerometers, magnetometers and rate gyros are used to feed an observer. The speed and quaternion estimation are used in a feedback loop. The reference corresponds to a quaternion (Fig. 2). For simplification,  $q=(1 \ 0 \ 0 \ 0)^T$  is chosen as reference for the following simulations.

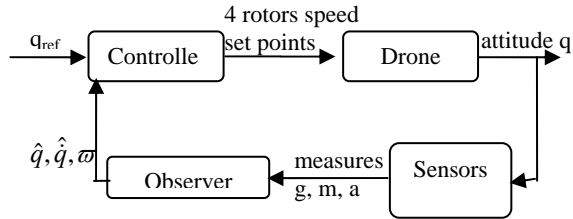


Fig. 2. The control loop.

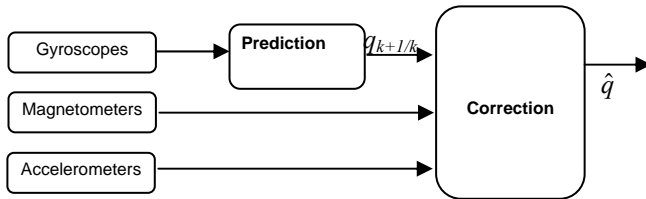


Fig. 3. The observer principle.

Several state observers have been tested. An extended Kalman filter (Fig. 3) has provided good results. In this filter, the prediction part of the algorithm uses only the gyros measurements while the accelerometers and magnetometers outputs are used for correction (Tanwani, 2006).

### 4. FDI ALGORITHM

As a state observer has been designed for the control synthesis, it seems quite natural to investigate if this observer is useful for diagnosis. The use of a bank of observer to design structured residuals is well known (Isermann, 2006).

#### 3.1. Diagnosis of Sensors

To isolate a fault concerning one of the output sensors, under the hypothesis that actuators are fault-free, a Generalized Observer Scheme (GOS) is used. Based on this principle, an observer that takes 8 out of 9 system outputs is used to generate structured residuals. The basic idea is to estimate a quaternion using these 8 sensor values and compare this to the quaternion calculated via the drone mechanical model (Fig. 4). The residual is thus insensitive to the faults in the sensor that is not used and the signature table strongly isolable.

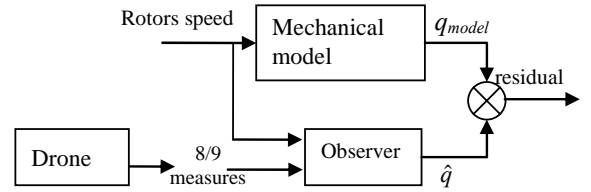


Fig. 4. Residuals generation using observers

Table 1 is the fault signature table obtained in case of faults in the three rate gyros. It shows that the estimator *gyro1* is sensitive to faults in all the sensors ( $f_{a1}, f_{m1}, f_{g1}, i \in 1, 2, 3$ ) except  $f_{g1}$ , which indicates the fault in rate gyro 1 (along x - axis). The same definition applies to all other symbols. Thus, it is possible to isolate the fault in rate gyros by looking at the residuals generated by the three observers mentioned in the first column of Table 1.

Table 1. Fault signature for diagnosis of rate gyros

	$f_{a1}$	$f_{a2}$	$f_{a3}$	$f_{m1}$	$f_{m2}$	$f_{m3}$	$f_{g1}$	$f_{g2}$	$f_{g3}$
<i>gyro1</i>	1	1	1	1	1	1	0	1	1
<i>gyro2</i>	1	1	1	1	1	1	1	0	1
<i>gyro3</i>	1	1	1	1	1	1	1	1	0

The same technique is applied for the surveillance of accelerometers and magnetometers but the values of these sensors are a nonlinear function of quaternion. The technique was modified and the observer was replaced by an optimisation algorithm to find an optimal quaternion  $q_{opt}$  (Fig.3). The technique of optimization would generate an optimal quaternion using five out of six measurements of accelerometers and magnetometers. Thus the idea of generalized observers is conserved but the observer is replaced by an estimator which uses optimization algorithm.

Table 2. Fault signature for diagnosis of accelerometers and magnetometers

	$f_{a1}$	$f_{a2}$	$f_{a3}$	$f_{m1}$	$f_{m2}$	$f_{m3}$	$f_{g1}$	$f_{g2}$	$f_{g3}$
<i>acc1</i>	0	1	1	1	1	1	0	0	0
<i>acc2</i>	1	0	1	1	1	1	0	0	0
<i>acc3</i>	1	1	0	1	1	1	0	0	0
<i>mag1</i>	1	1	1	0	1	1	0	0	0
<i>mag2</i>	1	1	1	1	0	1	0	0	0
<i>mag3</i>	1	1	1	1	1	0	0	0	0

Table 2 shows that the estimator *acc1* is insensitive to faults in accelerometer along x-axis ( $f_{a1}$ ). The '1's in the first row show that a non zero residual is generated by *acc1* if there appears a fault in any other accelerometer or magnetometer. Note that all these six estimators are insensitive to faults in rate gyros.

### 3.2. Algorithm

The observers that estimate the quaternion using 3 accelerometers, 3 magnetometers and 3 rate gyros already exist (Guerrero et al., 2005). The problem that confronts us is the quaternion estimation while neglecting one of the sensor values. In the formulation mentioned for rate gyros, it must be noted that 8 sensor measurements are used and the last one is modelled mathematically. The classical Kalman filter is further used to get  $\hat{q}$ . Residuals are generated following the scheme in Fig. 4. The first step is to find quaternion of error ( $q_e$ ) using:

$$q_e = q_{estim} \otimes q_{model}^{-1} \quad (15)$$

Since this is a quaternion of rotation, it has the following form:

$$q_e = \left[ q_{e0} \quad \bar{q}_e \right]^T = \left[ \cos\left(\frac{\phi_e}{2}\right) \quad u_e \sin\left(\frac{\phi_e}{2}\right) \right]^T \quad (16)$$

and finally the residual  $\phi_e$  is computed in degrees,

$$\phi_e = 2 \frac{180}{\pi} \arccos(q_{e0}) \quad (17)$$

The residual  $\phi_e$  actually has some physical meaning: it is the difference between the original orientation and the estimated orientation of the drone.

The above results have also been verified using a non linear observer for which the algorithm is given in the appendix.

While discarding one of the accelerometers or magnetometers measurements, these observers cannot be used to estimate the quaternion. Hence an optimization algorithm is proposed to calculate an optimal quaternion  $q_{opt}$ . The criterion to be minimized is:

$$f(q) = \sum_{j=1}^6 (C(q)\bar{r} - v_{mes}(j))^2 \quad (18)$$

where  $\bar{r} = \left[ \bar{g} \quad \bar{h}_m \right]^T \in \mathfrak{R}^{6 \times 1}$ . Once  $q_{opt}$  is calculated from this equation, the residuals can be computed in exactly the same way as mentioned in equations 15 through 17.

### 3.3. Diagnosis of Actuators

The 4 actuators are independent. The model of each electrical motor has been used for their surveillance. Due to the constraint imposed by the availability of a single sensor that measures the rotor speed, the principle shown in Fig. 5 is used for fault diagnosis of each rotor: Hence, the fault signature is given by Table 3.

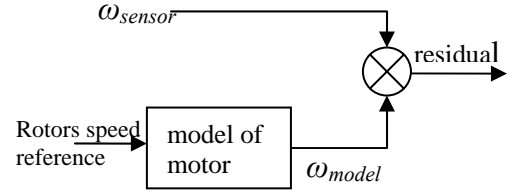


Fig. 5. Residual generation for actuators

Table 3 Fault signature for diagnosis of actuators

	$f_{rotor 1}$	$f_{rotor 2}$	$f_{rotor 3}$	$rotor 4$
<i>Rotor1</i>	1	0	0	0
<i>Rotor2</i>	0	1	0	0
<i>Rotor3</i>	0	0	1	0
<i>Rotor4</i>	0	0	0	1

There are several limitations to this technique. A fault in speed sensor or in motor cannot be differentiated. Also, the form of residual is strictly dependent upon the difference of speed which may not always be suitable for boolean decision making. This limitation could be removed by additional sensors to measure other variables such as current or voltage but it is contrary to the weight constraint.

### 3.4. Simulation results

The closed loop system in Fig. 2 is simulated with Matlab Simulink and various faults have been injected into the system. The FDI performance is now discussed.

In the first experiment (Fig. 6), the information delivered by the rate gyro along y-axis is assumed to be lost at  $t=5$  ( $f_{g2}$  fault, Table 1 and 2). The residuals generated by *acc1*, *acc2*, *acc3* (first row of Fig. 6) and *mag1*, *mag2*, *mag3* (second row) remain null even after the fault has occurred. Residuals generated by *gyro1* and *gyro3* are non zero after  $t=5$  and *gyro2* is insensitive to  $f_{g2}$  (third row). This is consistent with Tables 1 and 2.

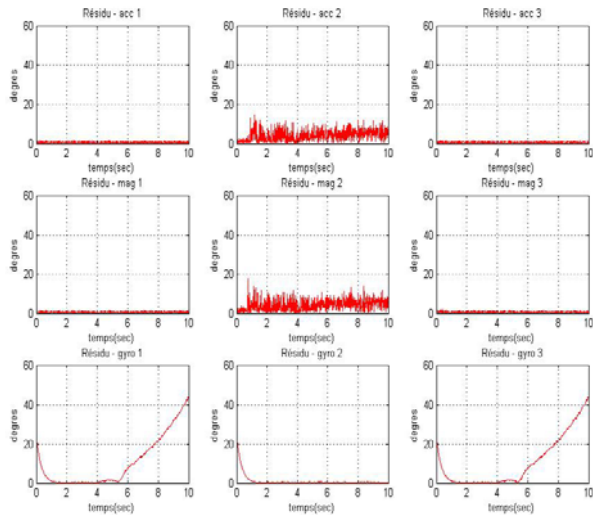


Fig. 6. Residuals generated by the 9 estimators for the total breakdown in rate gyro along y-axis ( $f_{g2}$ ).

In the second experiment, a failure of the magnetometer along x-axis is considered. The fault is simulated at  $t=4$ . In Fig. 7, the residuals generated by the six estimators of Table 2 are shown and once again the results are consistent with the theoretical formulation. Only estimator *mag1* is insensitive to the faults in the magnetometer along x-axis ( $f_{m1}$ ).

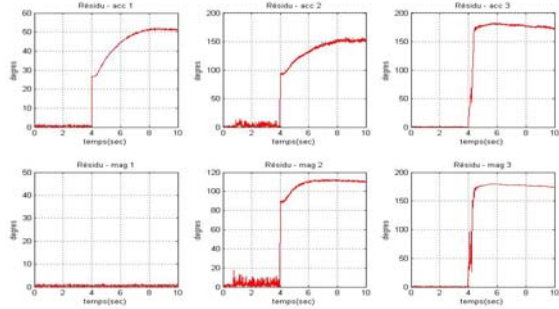


Fig. 7. Residuals generated by the 6 estimators of Table 2 for total breakdown in magnetometer along x-axis ( $f_{m1}$ ).

For actuators, Fig. 8 shows the case of a fault winding in rotor 1. The fault is simulated at  $t=5$  and the residual increases immediately.

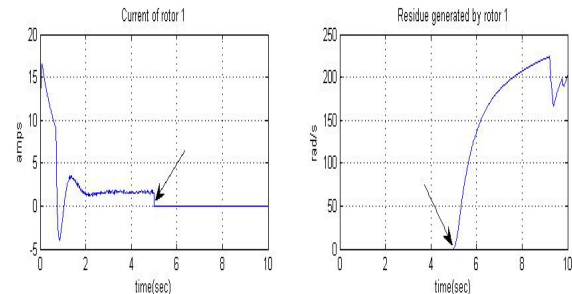


Figure 8. Fault winding in rotor 1.

## 5. NETWORK ARCHITECTURE

The benchmark presently in development will be equipped with two networks (Fig. 9).

The first network is an embedded network on the drone-helicopter itself. The control loop is closed through the embedded network that must be designed taking into account the fact that the system has fast dynamics and strong real-time constraints. The problems induced by the network are studied in (Tanwani et al., submitted).

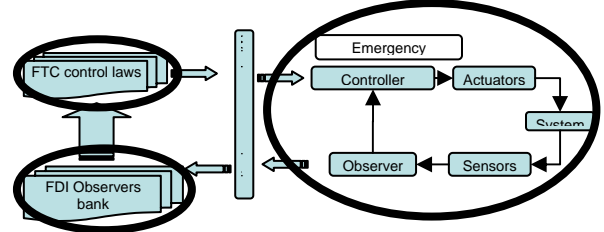


Fig. 9. The two networks.

The second network is a wireless network which aims at ensuring the communication between the drone-helicopter (a mobile vehicle) and the ground. It is evident that calculations related to the 9 observers and the 4 actuators fault detection algorithms can be really time-consuming. It is thus not reasonable to implement them on the embedded microcontroller. The FDI and FTC algorithms will be deported in the ground station. Evidently, the duration of the transmission will be longer, and the real-time constraints will not be met as easily as in the case of the wired network, so some strategies of Networked Fault-Tolerant Control and decentralised Diagnosis will be tested.

## 6. DISCUSSION

This paper has discussed the design of a benchmark embedded system: a drone, with state estimation and closed loop control. The original point of the presented approach is the use of a quaternion vector to represent the mobile position instead of the Euler angles. The quaternion is observed thanks to nine sensors. Some of these sensors can be related to the dynamic model while others represent static relations verified by the quaternion. Thus the observer equations are twofold: a least squares estimation algorithm for the static relations and an extended Kalman filter for the dynamic relations.

This observer needs to be modified to allow fault detection and isolation through a Generalized Observer Scheme. A strongly isolable signature table is obtained. Future work concerns the study of other types of observers that have been proposed in the literature for FDI. As a first principle reliable model of the process is available, approaches based on behavioural models such as neural networks do not seem, at a first glance, to be relevant (Koscielny et al., 2004). But approaches based on non linear observers require a lot of attention (Kinnaert, 1999; De Persis, 2001).

The residual generation and decision making process are time consuming. As time constraints are a crucial aspect in this application, attention must be paid to

the way algorithms can be implemented. It is hardly imaginable that the bank of observers can be executed in the embedded system. It is rather envisaged that the FDI algorithms will be deported on a ground station related to the drone by a wireless network. The embedded network will be in charge only of the closed loop control. The network effect on the residual evaluation will need a careful study. The idea for future research is to examine specific problems such as the network induced delays, the loss of messages (or packets) due to the worse Quality of Service between the drone and the ground station and the management of the roaming (between several ground stations).

## REFERENCES

- Chou, J. C. K. (1992). Quaternion kinematics and dynamic differential equations, *IEEE Transactions on Robotics and Automation*, **8**, pp. 53-64.
- De Persis, C., Isidori, A. (2001). A geometric approach to non linear fault detection and isolation, *IEEE TAC* 45(6), pp. 853-865.
- Fang, H., Ye, H., Zhong, M. (2006). Fault diagnosis of networked control systems. In: *IFAC Symposium Safeprocess 2006* (Beijing, PRC), pp 1-12.
- Guerrero Castellanos, J.F, Lesecq, S., Marchand, N., Delamare, J. (2005). A low-cost air data attitude heading reference system for the tourism airplane applications, *IEEE Sensors*, pp. 1388\_1391.
- Isermann R. (2006). *Fault Diagnosis Systems: An introduction from fault detection to fault tolerance*. Springer.
- Kinnaert, M. (1999). Robust fault detection based on observers for bilinear systems, *Automatica* 35(11), pp. 1829-1842.
- Koscielny, J., Kowalczyk, J. M., Cholewa, W., (2004). *Fault diagnosis: models, artificial intelligence, applications*. Springer-Verlag.
- Nilsson, J., Bernhardsson, B., Wittenmark, B. (1998). Stochastic analysis and control of real-time systems with random time delays, *Automatica* 34:1, pp. 57-64.
- Otanez, P. G., Moyne, J. R., Tilbury, D. M. (2002). Using Deadbands to Reduce Communication in Network Control Systems. In: *Proceedings of the American Control Conference ACC02*.
- Tanwani, A. K., Galdun, J., Thiriet, J-M., Lesecq, S., Gentil, S. Modelling of an experimental networked drone with consideration of faults, *submitted to ECC'07*.
- Tanwani, A. K. (2006). Stratégies de diagnostic pour un système embarqué: application à un mini drone 4 rotors, *Master Degree Report, Laboratoire d'Automatique de Grenoble* (Fr.), 43 p.
- Tayebi, P., Mc Gilvray S. (2004). Attitude stabilization of a four rotor aerial robot. In : *43<sup>rd</sup> Conference on Decision and Control*, Atlantis (Bah).
- Zhang, P., Ding, S. (2006). Observer-based monitoring of distributed networked control systems In: *IFAC Symposium SafeProcess 2006* (Beijing, PRC), pp.337-342.

## Appendix

### A. Linear Attitude Estimation

Let  $\vec{b}_i, i = 1 : n$ , be  $n$  vectors measured in  $B$  while  $\vec{r}_i, i = 1 : n$  are the corresponding vectors expressed in inertial frame  $R$ . Then, the measurement equation 12 and 14 at time  $t_k$  can be modeled as:

$$\vec{0} = \bar{H}q_k = [\varphi_{k,j}^T]q_k \in \mathfrak{R}^4$$

where  $\varphi_{k,j}^T \in \mathfrak{R}^4, j = 1 :$

$$4n, \bar{H}_k = [H_{k,1}^T \ H_{k,2}^T \ \dots \ H_{k,n}^T]^T \in \mathfrak{R}^{4n \times 4} \text{ and}$$

$$H_{..i} = \begin{pmatrix} 0 & -(\vec{b}_i - \vec{r}_i)^T \\ \vec{b}_i - \vec{r}_i & -(\vec{b}_i + \vec{r}_i)^{\times} \end{pmatrix} \in \mathfrak{R}^{4n \times 4}$$

Let  $\hat{q}_k$  be the attitude quaternion estimated at time  $t_k$ , then using the recursive least square,  $\hat{q}_k$  can be expressed as:

$$\hat{q}_{ms}(k) = \left[ \prod_{j=1}^{4n} \left( I_4 - \frac{\gamma \varphi_j \varphi_j^T}{\alpha + \varphi_j^T \varphi_j} \right) \right] \hat{q}_{ms}(k-1)$$

where  $\alpha \geq 0$  and  $0 \leq \gamma \leq 2$ .

For the present application,  $n = 2$ ,  $\vec{b}_1 = \vec{b}_{acc}$  and  $\vec{b}_2 = \vec{b}_{mag}$ . Correspondingly,  $\vec{r}_1 = \vec{g}$  and  $\vec{r}_2 = \vec{h}_m$  are the gravity and magnetic field vectors in the inertial coordinate system  $R$ .

### B. Non Linear Observer

A nonlinear observer is implemented jointly with the previous algorithm in order to estimate a reliable attitude quaternion:

$$\dot{\hat{q}}(t) = \frac{1}{2} \Xi(\hat{q}) [\omega_g - \hat{b} + K_1 \varepsilon]$$

$$\dot{\hat{b}} = -T^{-1} \hat{b} - K_2 \varepsilon$$

where  $T$  is a diagonal matrix of time constants and  $K_i, i = 1, 2$  are chosen to be positive definite matrices.

$\hat{q}(t)$  is predicted attitude at time  $t$ , obtained by propagating the kinematics equation using the measured angular velocity, the bias estimate and  $\varepsilon$  which depends on the discrepancy  $q_e$  between  $\hat{q}(t)$  and  $q_{ms}(t)$ .

$$q_e = \hat{q}(t) \otimes q_{ms}^{-1}(t) = [q_{e0} \ \vec{q}_e]^T$$

$$\varepsilon = \vec{q}_e \operatorname{sgn}(q_{e0})$$